Axel's algebraic, a take on 5D chess notation.

A. Introduction

Hello.

As many 2D chess regulars already know, 2Dchess has move notations, among which algebraic chess notation.

It's especially helpful to read, write or talk about a position, as well as to write, share and export games, even though the features are not supported yet.

So here is my stab at a 5D chess notation. It's a bit of a wall of text with few illustrations for now. If you want to learn by example, jump straight to the end of the document. (example section).

I first define an overly verbose notation, that I call Reversible Algebraic Notation (RAN), then I trim it to build Algebraic Notation (AN) which is hopefully readable.

The big upside of this notation is that it contains all the information needed. It thus is unambiguous and very simple to define since there are no disambiguation rules needed as in regular algebraic notation for 2D chess. The big downside is that it is overly verbose, and thus potentially harder to read.

It is however easy to derive many other notations, by just trimming unnecessary bits of information with reduction rules. That's how I've chosen to build a relatively concise Algebraic Notation (AN) for 5D chess, going from a very verbose notation and trimming it. It is in contrast to how algebraic notation is usually defined for 2D chess, with too few information and disambiguation rules

If you want to contribute feel free to copy and branch this document, or to suggest changes or ask permission to edit via comment or discord (@4xel).

Disclaimer :I realize it was a bit preposterous to name my notation simply The Algebraic notation given the dev already has a home brew notation and many other community member are coming up with their own, and the community will probably eventually settle down on

something that isn't exactly any of the current existing languages. Outside the scope of this doc, feel free to refer to my notations however you may like (4xel's notation might be an option, simply AN or RAN as well if not ambiguous).

B. Reversible Algebraic Notation (RAN)

1. General structure

5D chess still is a turned based game, but unlike 2D chess, you may be allowed or required to make several moves within each turn. The notation structure reflects this the following way:

<ply (half turn) prefix>.
    <move 1>;
    <move 2>;
    <move 3>.

I chose the punctuation (';', and '.') according to my preference and to make spacing and line break undoubtedly optional, but I don't know if punctuation is actually necessary (depending on how moves and prefix are encoded).

2. Turn prefix

<whose turn it is> <turn number> <time of the present>
In the format (ignoring whitespaces):
`regex("([bw])(\d+)T(\d+)\.")`

Examples:
b24 T13.
means it's black's turn,it's turn 24, the present is at time 13.
For comparison, 2D chess turn prefix does not include the time of the present, and indicates whose turn by a terminal dot ('.', White) or ellipsis ('...', Black).

3. Square location

A square location through space and time is entirely defined by the following 5 dimensions:

- Whose turn it is.
- Which timeline it belongs.
- When in time.

- Its file.
- Its rank.

It can be given in the following format, consistent with in game notations:
`regex("([bw])L([+-]?\d*)T(\d+)([a-h])([1-8])")`

The central timeline, if unique, can be referred to as L0 or L. If there are two central timelines, they are L-0 and L+0, the signs being mandatory.

Example:
w L-4 T15 d5
Means: white's turn, timeline -4 (the fourth Black assuming initial position has 2 timelines or less), time 15, rank d, file 5.

As far as move notation is concerned, whose turn it is is unnecessary when it is already contained in the turn prefix, and a piece of a given color can only go to a board when it is or was its own color turns. Which simplifies the format to:
`sq = regex("L([+-]?\d*)T(\d+)([a-h])([1-8])")`
example :
L T12 f6
Means: timeline 0 (unique 0-th timeline) time 12, rank f, file 6.

4. Move notation

Much like in RAN for 2D chess, a move is done by specifying the piece moving, the initial square and the target square.

Caution: the destination is the target square, the one that you click, not the one at which the piece first appears (which is half a time increment later and possibly off to some branch).

Pieces are denoted by their first letter in upper case, with exception of kNight.
[PNRBUDKQ]
P is for pawn, U for unicorn, D for dragon
The piece is inserted between it's time and space position, the idea being to write in order of relevance (you first look for a board with time coordinates, then a piece, then confirm with spatial coordinates).

Example:
L+1T23 Ug4 L-0T21 e4
means :
timeline +1, time 23, Unicorn from file g rank 4 goes to
timeline -0 (where timeline +0 exist and is distinct), time 21, file e, rank 4

1. Capture, promotions

‘x’ followed by the nature of the captured piece is inserted
Example:
L0T6Nf3 xP L0T6e5
Means timeline 0 time 6, night from f3 takes pawn from timeline 0 time 6 at e5.

Promotions are indicated as in regular chess, and indicating them is currently optional as underpromotion isn’t a thing.
Example:

2. Branching

When a branch is created, additional information is necessary, much like upon promotion. It is done within parentheses according to the following format:

regex (“\(\+L([+-]?\d*)\)”)
Example:
L0T6 Ng1 L0T5g3 (+L3)
From timeline 0 time 6, kNight g1 goes to timeline 0 time 5, g3, creating timeline 3.

3. Additional info

Useful additional info can be added much like for 2D chess. Worthy of mention are forcing moves, for which we can be a bit fancier:

- + : check to an active King
- * : check to a history King (one which cannot move)
- * + : check an active King and a history King
- ++ checkmate on a board (check from a board and no single move from that board to itself can save you)
- +* stalemate on a board (no legal move from that board to itself)
- # global checkmate

Preferably, a check should be signaled after the move that triggers it,after the board (timeline and time) when the threatening piece is at.

A ‘p’ following a newly created timeline can be used to indicate that it is passive.

C. Coordinate Notation

As an example to how to derive other notations from the very thorough algebraic reversible. Coordinate notation is like RAN with the following differences:

- No present time in the Ply Prefix

- Moves are simply noted with initial square and target square

Nothing more is required, if one has knowledge of the whole game position, the new position after given moves can be inferred.

Given how unfriendly coordinate notation feels to humans for 2D chess, I don't have my hopes high for it in 5D, but like 2D, it might be useful to some engines.

D. Concise Reversible Algebraic Notation (CRAN)

Could be done and could have some merit, but for a game where a position contains most of the history of the game (not quite), it seems a bit moot to take reversibility as the criteria for which piece of information can be ditched and which should be held.

Hence I jump straight onto Algebraic Notation. In practice, if you ever see me using notation outside of this document, it will probably be an ill defined hybrid between AN and RAN.

E. Algebraic Notation (AN)

As you may have noticed, RAN takes a lot of space unnecessarily, for example "timeline 0 time 6, night from f3 takes pawn from timeline 0 time 6 at e5" could have been stated: "timeline 0 time 6, night from f3 takes pawn e5"

AN is defined from RAN with the following differences.
For ply prefix :

- **Whose turn it is optional** (still recommended after long plies)
- **Time of the present** is optional if the present flew naturally (½ unit of time) from last ply. Still recommended if knowing that time is relevant for some moves of the ply.

For moves:

1. **The timeline of an initial square** is not given if it's the next active board, in ascending order for white, descending from black, starting from the first. In other words, moves are assumed to be given from top to bottom board from the player respective perspective, unless specified otherwise.
2. **The time of an initial square** is not given. Moves can only be done from an active board, so the time of a move can be deduced from its timeline (each timeline only has one active board).
3. **The rank of an initial square** is omitted if not needed to identify the piece.
4. **The file of an initial square** is omitted if not needed to identify the piece (after accounting for potential rank omission).

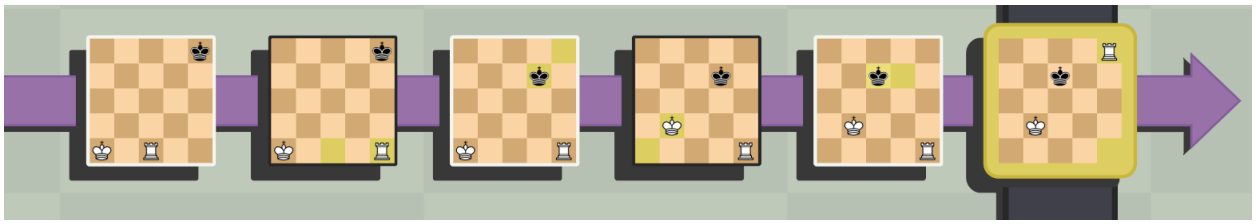
5. **Both rank and file of a target square** are simultaneously omitted if both identical to initial square's (pure time travel).
6. **A time variable of a target square** (timeline or time) is omitted if it is the same as the initial square's.
7. **The name of pieces taken is omitted**.
8. **The name of pawns is omitted**.

3 and 4 is the 2D AN way of reducing information, whereas 6 is the 2D CRAN way.

F. Examples

Beware, contains puzzle spoilers.

1. Rook tactic I



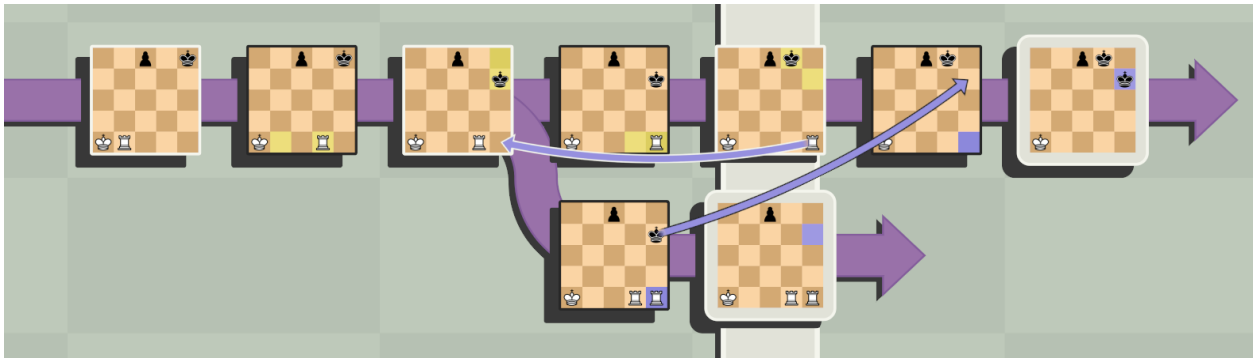
- RAN :

w3T3. LT3Re1 LT3e5* #.
Alternatively, we can count from turn 1 for a puzzle rather than trying to infer how many turns were played (w1T3 instead of w3T3).

- AN :

1. Re5* #.

2. Rook tactic II



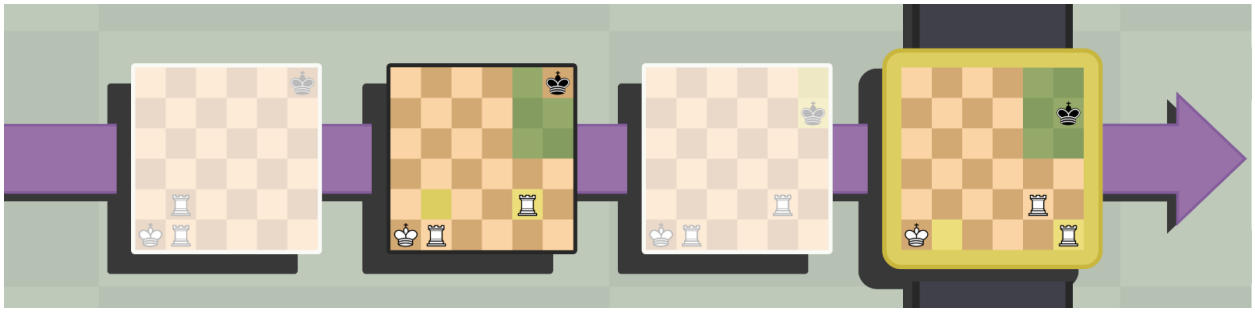
- RAN :

w1T3. L+1T3Rd1 L+1T3d5* #.

- AN :

1. L1Rd5* #.

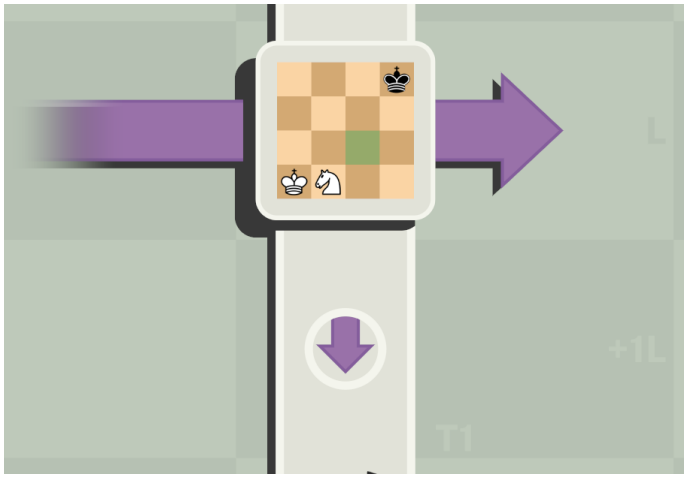
3. Rook tactic III



- AN :

1T2. Re1++ .. K T1e3 (+L-1).
2T1. R e1++ #.

4. Knight tactic I

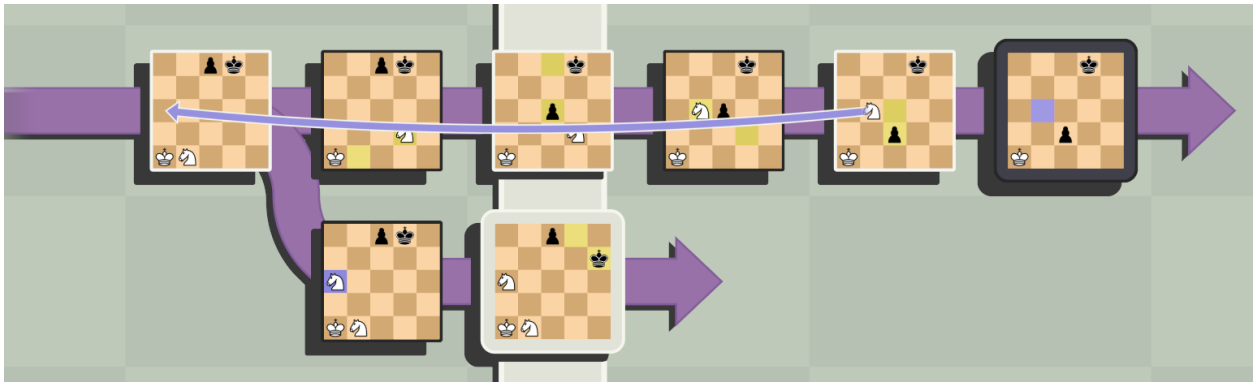


Note the very faint time coordinate (L +1L, T1)

- AN :

1.Nd2* #

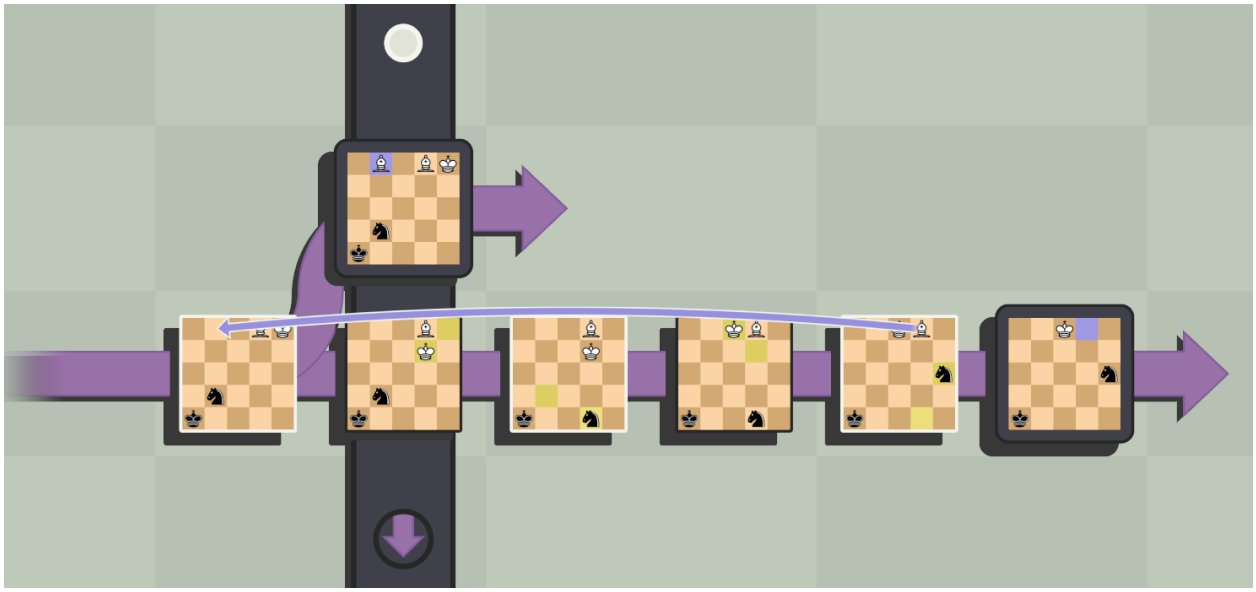
5. Knight tactic III



- AN :

1.Nb5* #
The timeline does not need to be given: L0 is active, but not for white.

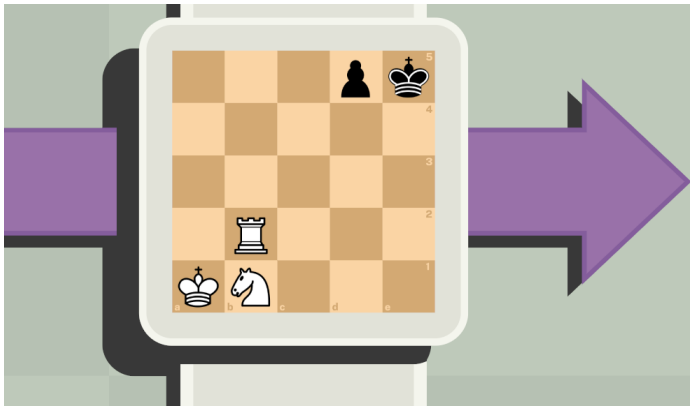
6. Knight tactic VII



- AN :

1T3.L0N L-1T1* #
Pure time travel, no need to give space coordinates.
By the way some of my diagrams do not have enough information to know the name of the timeline (they are -1, 0 and 1). In addition or replacement to zooming out to see them written very faintly, one could also rely on the color of the timelines (purple = active, white = inactive) and on the indicator on the present line. Here for example, you can see that Black can create another active timeline whereas White can't.

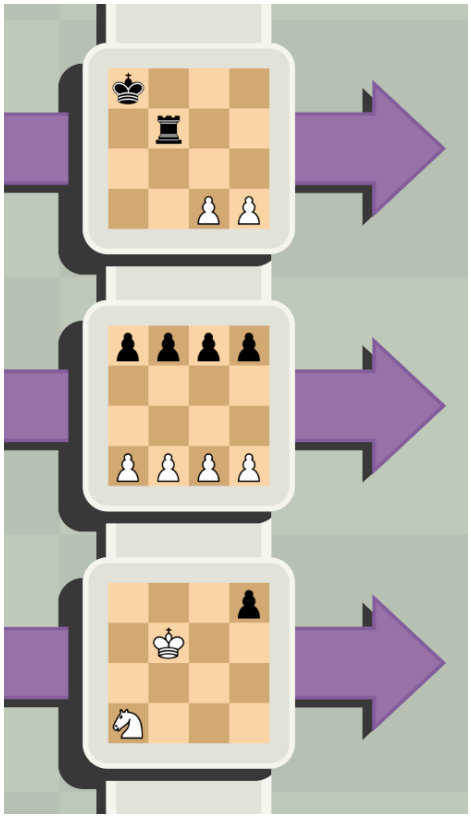
7. Combination Attack I



- AN :

1. Re2+ .. Kd4.
2. Nd2* #

8. Combination Attack II



- AN :

1. L+1N L-1a2 *; b3 #

Remarks:

- * is put at the end of a move whereas # is put at the end of a turn.
- Indicating the departing timeline is necessary here.